WHAT IS CLAIMED IS:

CLAIMS

1. A method for managing memory, comprising:

    maintaining a memory pool;

    specifying a specified amount of memory in the memory pool for

allocation; and

    requesting a process to release a requested amount of memory in the

memory pool.

2. A method for managing memory as recited in Claim 1, wherein the process

operates in a garbage-collected environment.

3. A method for managing memory as recited in Claim 1, wherein the process is a

Java process.

4. A method for managing memory as recited in Claim 1, wherein the process is a

Java program.

5. A method for managing memory as recited in Claim 1, wherein the process is a

memory-releasing process; and further comprising allocating the specified amount

of memory to a memory-requesting process.

6. A method for managing memory as recited in Claim 1, wherein the memory pool

includes reserved memory.

7. A method for managing memory as recited in Claim 1, wherein the memory pool

includes memory owned by a plurality of processes.

8. A method for managing memory as recited in Claim 1, wherein the memory pool

includes a plurality of subpools.

9.   A method for managing memory as recited in Claim 1, further comprising determining that the specified amount of memory is required for allocation.

10.   A method for managing memory as recited in Claim 1, further comprising receiving a memory request from a requesting process for the specified amount of memory and determining that the specified amount of memory is required for allocation.

11.   A method for managing memory as recited in Claim 1, wherein the process is a releasing process; and further comprising monitoring a monitored process and determining that the monitored process requires additional memory.

12.   A method for managing memory as recited in Claim 1, wherein the process is a releasing process; and further comprising monitoring a monitored process and detecting a rate of garbage collection for the monitored process.

13.   A method for managing memory as recited in Claim 1, wherein requesting a process to release a requested amount of memory in the memory pool includes selecting the process from a plurality of processes based on status information.

14.   A method for managing memory as recited in Claim 1, wherein requesting a process to release a requested amount of memory in the memory pool includes making a request via a system call.

15.   A method for managing memory as recited in Claim 1, wherein requesting a process to release a requested amount of memory in the memory pool includes making a request via an inter-process communication protocol.

16.   A method for managing memory as recited in Claim 1, wherein the specified amount of memory is approximately equal to the requested amount of memory.

17.   A method for managing memory as recited in Claim 1, wherein the requested amount of memory in the memory pool is freeable memory.

18.   A method for managing memory as recited in Claim 1, further comprising refilling a subpool of the memory pool with the requested amount of memory released by the process.

19.   A computer program product for managing memory, the computer program product being embodied in a computer readable medium and comprising computer instructions for:

   maintaining a memory pool;

   specifying a specified amount of memory in the memory pool for allocation; and

   requesting a process to release a requested amount of memory in the memory pool.

20.   A memory management system, comprising:

   a memory pool;

   a processor coupled to the memory pool, configured to:

      maintain the memory pool;

      specifying a specified amount of memory in the memory pool for allocation; and

      request a process to release a requested amount of memory in the memory pool.

21.   A method for managing memory, comprising:

   maintaining a memory pool;

receiving status information from a plurality of processes; and

managing memory among the plurality of processes using the status

information.

22.   A method for managing memory as recited in Claim 21, wherein managing

5   memory among the plurality of processes includes allocating memory to one of

the plurality of processes.

23.   A method for managing memory as recited in Claim 21, wherein managing

memory among the plurality of processes includes requesting one of the plurality

of processes to release memory.

10   24.   A method for managing memory as recited in Claim 21, wherein the status

information includes status of freeable memory.

25.   A method for managing memory as recited in Claim 21, wherein the status

information includes efficiency of the process's garbage collector.

26.   A method for managing memory as recited in Claim 21, further comprising

15   requesting status information.

27.   A method for managing memory as recited in Claim 21, wherein the status

information is sent along with a memory allocation request.

28.   A method for managing memory as recited in Claim 21, wherein the status

information is received periodically.

20   29.   A computer program product for managing memory, the computer program

product being embodied in a computer readable medium and comprising

computer instructions for:

maintaining a memory pool;

receiving status information from a plurality of processes; and

managing memory among the plurality of processes using the status

information.

30.   A memory management system, comprising:

5   a memory pool;

a processor coupled to the memory pool, configured to:

maintain a memory pool;

receive status information from a plurality of processes; and

manage memory among the plurality of processes using the status

10   information.

31.   A method for managing memory, comprising:

maintaining a memory pool;

determining that an amount of memory in the memory pool is required for

allocation; and

15   allocating the amount of memory to a process.

32.   A method for managing memory as recited in Claim 31, wherein determining that

an amount of memory in the memory pool is required for allocation includes

determining a request priority.

33.   A method for managing memory as recited in Claim 31, wherein determining that

20   an amount of memory in the memory pool is required for allocation includes

receiving a request having an urgency level and determining a request priority

based on the urgency level.

34.   A method for managing memory as recited in Claim 31, wherein determining that

an amount of memory in the memory pool is required for allocation includes

determining whether the process includes memory collateral.

35.   A method for managing memory as recited in Claim 31, wherein allocating the

amount of memory to a process includes selecting the amount of memory from a

subpool of the memory pool.

36.   A method for managing memory as recited in Claim 31, wherein allocating the

amount of memory to a process includes selecting the amount of memory from a

subpool of the memory pool; further comprising refilling the subpool with

released memory by the process.

37.   A computer program product for managing memory, the computer program

product being embodied in a computer readable medium and comprising

computer instructions for:

    maintaining a memory pool;

    determining that an amount of memory in the memory pool is required for

allocation; and

    allocating the amount of memory to a process.

38.   A memory management system, comprising:

    a memory pool;

    a processor coupled to the memory pool, configured to:

        maintain a memory pool;

        determine that an amount of memory in the memory pool is

required for allocation; and

allocate the amount of memory to a process.

5